Aaron Schendel

Dr. Zelle

CS 230

20 October 2011

Case Study – C#

Language Environment

When one decides that they want to start learning about and/or programming in the language C#, or any .NET language, they realize that the .NET framework was created specifically for Windows systems. While this is nice for Windows users, it leaves all the Mac OS and Linux users out there in the dark. This is why a group of software engineers, who later started the company Xamarin, decided to make a cross-platform implementation of Microsoft .NET called Mono. In other words the .NET framework is for Windows only and Mono is for any of the heavy hitters: Windows, Mac OS, and Linux. On the Mono project website Mono is defined as “Mono is a software platform designed to allow developers to easily create cross platform applications. It is an open source implementation of Microsoft's .Net Framework based on the ECMA standards for C# and the Common Language Runtime.” They later state that in continuing the development of this implementation “we can lower the barriers to producing great applications for Linux.” So clearly, I chose to use Mono.

The programming environment that I chose to use is a package that is included in the current (11.10) version of Ubuntu called MonoDevelop. MonoDevelop is defined on its project website as “MonoDevelop is a free GNOME IDE primarily designed for C# and other .NET languages, although open to any kind of language. However, MonoDevelop hopes to be more than just an IDE: it intends to be an extensible platform upon which any kind of development tool can be built.” Note that this IDE can be used on Windows, Mac OS, and many forms of Linux; I just prefer to use the Ubuntu distribution of Linux. For Ubuntu, obtaining Mono and MonoDevelop is as simple as opening the Ubuntu Software Centre and searching for MonoDevelop and clicking download. You don't have to worry about installing the Mono compiler itself because it comes already installed in the Ubuntu distribution of Linux.

This programming environment seems like a stripped down version of Eclipse, but still very versatile and useful. To create a simple C# program one needs to first start up MonoDevelop and then in the top left there is an icon to create a new file. A menu comes up that allows one to choose between various .NET languages including C, Objective C, VBNet, and of course C#. Once C# is clicked on it gives various options but for our purpose we click on General. We then have the option to create a new file with an already created class, enumeration, interface, or structure. There is also an option to create an empty file with nothing created for the user. For creating a simple “Hello, World” function we are going to open a new file with a class already created for us. We won't go in to the actual coding yet, but once the code is typed into the class we can then check to see if it works. First we save the file by pressing Ctrl-S, and then we build the program by hitting F7, followed by actually running the program by pressing Ctrl-F5. Note that there are actual buttons that can be clicked on in the toolbar at the top of MonoDevelop; however the shortcuts save precious coding time. If there is an error in our code then we won't be able to get past the build step and in the body of the program it will show where the error is and it will also show on the bottom left a “Build Error” tab that when opened shows how many errors you have and tells where they are.

In summation, the form of C# that I have chosen to use is called Mono and the reason I chose it is because of its cross-platform versatility. The programming environment that is being used is called MonoDevelop and it was chosen because it was created by the developers of Mono and it has a very nice structure with many programming tools built into it, as well as it is extremely easy to obtain when using Ubuntu.

Bibliography MLA

Mono Development Team. "What Is Mono." *Mono*. Xamarin. Web. 20 Oct. 2011. <<http://www.mono-project.com/What_is_Mono>>.

Mono Development Team. "General Questions." *MonoDevelop*. Xamarin. Web. 20 Oct. 2011. <http://monodevelop.com/>.

Sebesta, Robert. *Concepts of Programming Languages*. Ninth. Pearson Education, Inc., 2009. 103-4. Print.